7.REFERENCES

[1] C.H Yun and M.S. Chen, "Mining Web Transaction Patterns in an Electronic Commerce Environment", in Proceedings of the 4th Pacific-Asia Conf. on Knowledge Discovery and Data Mining, April 2000.

[2] D. Hawking, E. Voorhees, P. Bailey, and N. Craswell, "Overview of TREC-8 Web Track", In Proceedings of TREC-8, pp. 131–150, Gaithersburg MD, November 1999.

[3] DirectHit: http://www.directhit.com.

[4] H. Mannila, H. Toivonen, and A. I. Verkamo, "Discovery of frequent episodes in event sequences", Data Mining and Knowledge Discovery, 1(3): 259 - 289, November 1997.

[5] J. Han, Jian Pei, and Yiwen Yin, "Mining Frequent Patterns without Candidate Generation", ACM SIGMOD Intl. Conference on Management of Data, 2000.

[6] J. Kleinberg, "Authoritative Sources in Hyperlinked Environment", in Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithm, 1998.

[7] J. Pei, J. Han, B. Mortazavi-Asl, and H. Zhu, "Mining Access Pattern efficiently from web logs", in Proceedings of 2000 Pacific-Asia Conf. on Knowledge Discovery and Data Mining, April 2000.

[8] M. Chen, J. Park, and P. S. Yu, "Efficient Data Mining for Path Traversal Patterns", IEEE Trans. on Knowledge and Data Engineering, Vol. 10, No. 2, pp. 209-221, April 1998.

[9] M. Hearst, "Next Generation Web Search: Setting Our Sites", IEEE Data Engineering Bulletin, Special issue on Next Generation Web Search, Luis Gravano(Ed.), September 2000.

[10] M. Spiliopoulou and C. Pohle, "Data mining for measuring and improving the success of Web sites", Data Mining and Knowledge Discovery, 5:85–14, 2001.

[11] M. Spiliopoulou, C. Pohle, and L. Faulstich, "Improving the effectiveness of a Web site with Web usage mining", In Advances in Web Usage Analysis and User Profiling, Berlin, Springer, pp. 142–162, 2000.

[12] M.Chen, M.Hearst, J. Hong, and J.Lin, "Cha-Cha: A System for Organizing Intranet Search Results", In Proceedings of the 2nd USITS, Boulder, CO, October 1999.

[13] M.Levene and R.Wheeldon, "A Web Site Navigation Engine", in Proceedings 10th International WWW Conference, 2001.

[14] P. Hagen, H. Manning, and Y. Paul, "Must search stink? The Forrester report", Forrester, June 2000.

[15] Q. Yang, H. Hanning Zhang, and I.Tianyi Li, "Mining Web Logs for Prediction Models in WWW Caching and Prefetching", In The Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD'01, Industry Applications Track, August 2001.

[16] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules", in Proceedings of the 20th Int'l Conference on VLDB, Santiago, Chile, September 1994.

[17] R. Agrawal, T. Imielinski, and A. Swami, "Mining Associations between Sets of Items in Massive Databases", in Proceedings of the ACM-SIGMOD 1993 Int'l Conference on Management of Data, Washington D.C., May 1993.

[18] R. Baeza-Yates and B.Ribeiro-Neto, *Modern Information Retrieval*, Addison-Wesley, 1999.

[19] R. Cooley, B. Mobasher, and J. Srivastava, "Data Preparation for Mining World Wide Web Browsing Patterns", Knowledge and Information Systems V1(1), 1999.

[20] R. Cooley, P. Ning Tan, and J. Srivastava, "Discovery of Interesting Usage Patterns from Web Data", To appear in Springer-Verlag LNCS/LNAI series, 2000.

[21] R. Kosala and H. Blockeel, "Web Mining Research: A Survey", ACM SIGKDD, July 2000.

[22] R. Srikant and R. Agrawal, "Mining Generalized Association Rules", in Proceedings of the 21st Int'l Conference on Very Large Databases, Zurich, Switzerland, September 1995.

[23] S.Brin and L. Page, "the Anatomy of a Large-Scale Hypertextual Web Search Engine", in Proceedings 7th International WWW Conference, 1998.

[24] T. Cover and J. Thomas, "Elements of Information Theory", John Wiley & Sons, 1991.

[25] W. Lin, S. Alvarez, and C. Ruiz, "Efficient Adaptive-Support Association Rule Mining for Recommender Systems", Data Mining and Knowledge Discovery, 6, pp. 83–105, 2002.

[26] Z. Albrecht and A. Nicholson, "Predicting users' requests on the WWW", in J Kay (ed), CISM Courses and Lectures No. 407, International Centre for Mechanical Sciences, Proceedings of the Seventh International Conference on User Modeling, Banff, Canada, June 1999.

following is the result of our experiments. And the remark in Figure 7 is the topic of the real website.

The mining result is then used to improve the performance of the site search according to algorithms described in section 4.5.

Table 3. Results of Association Rule

Association Rule	Supp.	Conf.
~qtluong/gallery/slides-ice/folio-ice2b.html => ~qtluong/gallery/index.html	5.0%	83%
~luca/cs170/project/ex1.dat => ~luca/cs170/index.html	3.2%	93%
~gnguyen/ns/navbtn.htm ~gnguyen/ns/outlinec.htm => ~gnguyen/ns/img001.htm	2.4%	90%

Table 4. Results of Generalized Association Rule

#	Generalized Association Rule	Supp.	Conf.	Remark
1	~adj ~christos people/faculty/homepages ~bh => ~ddgarcia	0.6%	98%	Computer vision
2	~eanders/pictures ~qtluong/landscapes/lf ~qtluong/photography/lf => ~jhauser/pictures/history	0.6%	87%	Picture
3	~harrison ~hilfingr ~yelick people/faculty/homepages=> ~jordan	0.6%	97%	Machine Learning
4	~yuhong ~yinli ~xiaoye => ~xia	0.3%	100.0%	Chinese Staff
5	~wilensky/cs188/lectures ~xuanlong/cs188=> ~wilensky/cs188/assignments	0.3%	95%	Same Course

Our algorithm is compared with the pure text-based site search method, DirectHit algorithm, association rule and generalized association rule. The 4 algorithms are run on each of the query and the precision of results is evaluated. Several volunteers are required to do tests on our platform to pose the following ten queries and evaluate the relevance of searching results. We then computed precision of top 20 pages for each algorithm-query pair. Following is our queries: *hyperlink, EM, object oriented program, reinforcement learning, vision, Bayesian network, HMM, data mining, picture, Jordan*. The result is shown in Figure 6.

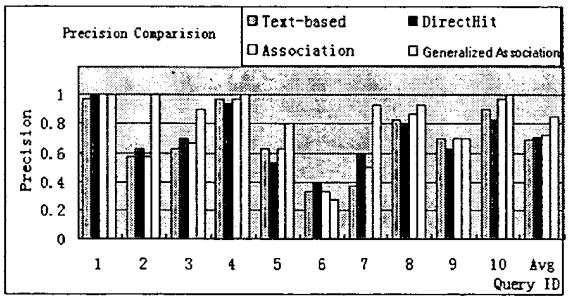

Figure 6. Pr cision comparison for 4 algorithms

In the experiment, we set the parameter α to be 0.7, The comparison of precision for 4 algorithm is shown in Figure 6. The result labeled as Avg is the average of all of the above 10 queries. According to Avg, we found that our proposed algorithm outperform the "full text search" and DirectHit algorithm. The average improvement of precision over the "full text search" is 15% and 13% for DirectHit.

As can be see from Figure 6, standard association rules don't improve the search result significantly. Based on our analysis, we find that few standard association rules contain the pages which are both in query result because of the diversity of the users' access behaviors and complexity of the website. With the help of generalized association rule, some associated topics can be discovered. In a topic, there are many pages which are similar in content. Therefore the ratio of co-occurrence of the pages in the query result is increased. Hence it can improve the performance of site search. DirectHit incorporates the frequency of user's access to compute the page score. The higher frequency a Web page is visited by user, the more important the Web page is. However, according to our statistics, in most of cases, the user only clicks the first 1-10 pages which are ranked by the page score. Therefore, DirectHit can not make significant improvement for the site search.

6.CONCLUSIONS AND FUTURE WORK

This paper discusses a log mining method for improving search functionality inside the website. We propose a generalized association rule mining method, which utilizes a taxonomy of website to mine for association rules at different levels. We also propose a novel re-ranking method using these generalized association rules. Our experiments show that the method is efficient and feasible for site search.

The construction of the taxonomy is based on the URLs of the pages, which implies that the underlying page organization reflects the semantics of the pages. In case this cannot be assumed, the taxonomy should be constructed according to the semantics of the pages, e.g., using content-based hierarchical clustering method, which is one of our on-going works.

Our algorithm use generalized association rules to re-rank Web pages based on previous pages in the same user session. This search scheme is much restrictive in some cases. Intuitively, association rules among Web pages can be seen as additional hyperlinks (or implicit links) among Web pages. Thus they could be utilized in link analysis algorithm such as HITS and PageRank. How to calculate this kind of implicit links, combine implicit links and explicit hyperlinks together, and apply them to link analysis algorithms, are main lines of our future researches.

make use of the previous users' query sessions. DirectHit algorithm uses the click popularity to improve the performance of the search. The higher frequency a Web page is visited by user, the more important the Web page is. Compared to DirectHit, our algorithm utilizes the popular clicked pages of the same query and improves the associate pages' rank which is based on the association rules.

First we implement the DirectHit algorithm on a "full text search" engine. DirectHit uses the sessions of the users' queries and the pages which are relative to the users' queries.

1. Through the site search engine, the user inputs a query word Q, then the search engine returns a result set D with the score which is based on the similarity of the page d and the query Q.
2. The pages are re-ranked according to the similarity score and the click popularity;

$$Score(d) = \alpha \times Sim(d) + (1-\alpha) \times Pop(d)$$

where Sim (d) is the similarity between the query Q and the page d, the Pop (d) is the click popularity of the page d.

Then we implement our algorithm using generalized association rules.

1. The user inputs a query word Q, and get a result set D with the score which is based on the similarity between the page and the query.
2. Similar to the pseudo relevance feedback approach, we get top n pages as a set P from the returned result.
3. Then we get the rules whose antecedents contain the pages in P or the ancestors of the pages in P, and acquire the descendant of the rules as a set R.
4. According to R, we calculate the support and the confidence of each page d in the result set D, if the page d is in R, we calculate the support and the confidence of the page d directly, if a parent node of d is in R, we calculate the support and the confidence of the page d according to the ratio of the page to its parent.
5. The result is re-ranked according to the similarity, the support and the confidence;

$$Score(d) = \alpha \times Sim(d) + (1-\alpha) \times Supp(d) \times Conf(d)$$

where $Sim(d)$ is the similarity between the query Q and the page d. $Supp$(d) and $Conf(d)$ are support and confidence of the rule of page d respectively.

5.EXPERIMENTAL RESULTS

In order to test the effectiveness of our proposed algorithm, Berkeley CS website is used as the experiment. We downloaded one-month's log file from http://www.cs.berkeley.edu/logs. The log records users' visit information, in which one record is corresponded to one HTTP request for a Web object by a specific user. After preprocessing, only text pages are reserved in the final dataset, which contains 112,059 pages, 296,667 users and 1,474,389 visit records. Figure 5 shows a statistic of frequency distribution of the Web pages.

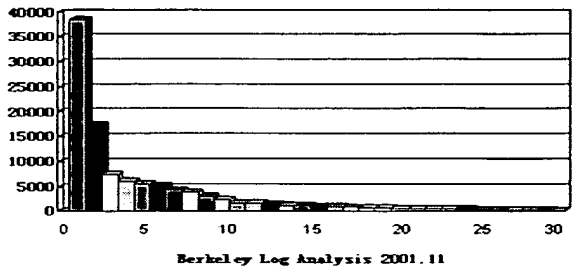

Figure 5. Frequency distribution of the pages. X axis: the hits number of a page, Y axis: the page numbers

In our experiment, we mine for generalized association rules with confidence of at least 60% and support of at least 10 occurrences in transactions.

To compare generalized association rule mining with standard association rule mining, we also conduct a standard association rule mining experiment without taxonomy. Mining for generalized association rules took about 19.7 minutes while mining for standard association rules took about 11.4 minutes. The rule statistics are as Table 2.

Table 2. Rule statistics

Methods	Rules Mined	Pruning		Summary	Rules Left
		Uninteresting	MI		
Generalized	246538	3042	32204	130024	146280
Standard	101304	0	16537	49832	59848

Observe that although the number generalized association rules is significantly more than stand association rules. A greater percentage of those rules are eliminated during pruning.

After pruning and summarization, we find something very interesting in the result that most items in the standard association rules have hyperlink relationships. However the generalized association rule mining can discover not only standard rules, but also high level relationships, which can not be explored by hyperlink structure. For example, In Table 4, rule 1 shows that ~adj, ~bh and ~ddgarcia are the pages, which are pertaining to the staff who work on computer vision. And we can also discover the rules, which can find those pages being of the same interest and about the same courses. The

various stages in the algorithm and the number of ancestors added to form extended transaction.

The performance of the algorithm *apriori* is poor, because there exist at least 10^5 items, and millions of candidate itemsets being created in multiple scans of the database. So we designed a generalized version of FP-growth algorithm [5].

The basic algorithm of generalized FP-growth has three steps: (1) generalized FP-tree construction, (2) generalized FP-tree generation and (3) optimization. The first step and the second step are same as described in [5], the difference is that transactions are replaced by extended transactions which include the parent nodes of original transaction nodes. The third step is the optimization of algorithm according to the following property [22]:

> The support of an itemset X that contain both an item x and its ancestor x' will be the same as the support of the itemset $X-\{x'\}$.

After the frequent itemsets have been created, the algorithm generates the association rule as described in [17];

4.4.Pruning and Summarization

We do not present all generalized association rules but only those that we deem "interesting". This is particularly important since, in practice, we find that there are thousands of similar and/or redundant rules that would damage the search performance otherwise.

Uninteresting Rules Pruning: Suppose we have two association rules: $X \Rightarrow Y$ and $X' \Rightarrow Y$, where X' is a parent of X. We define the rule $X \Rightarrow Y$ is *uninteresting*, iff

$\text{Support}(X \Rightarrow Y) / \text{Support}(X' \Rightarrow Y) \approx \text{Support}(X) / \text{Support}(X')$

#	Rule	Supp.
1	Outerwear ⇒ Footwear	8
2	Jackets ⇒ Footwear	4

Item	Supp.
Outerwear	2
Jackets	1

Figure 4. Examples of interesting and uninteresting rule

The proof is given by [22]. As the example shown in Figure 4, assuming we have the same taxonomy as in Figure 1, we do not consider rule 2 to be interesting since its support can be predicted based on rule 1. In other words, we can see from the right table that just 1/2 of amount of buying Outerwear are buying Jackets. Afterward, we can calculate the support of Jackets⇒Footwear to be just 1/2 of the support of Outerwear⇒Footwear. Thus the rule Jackets⇒Footwear is uninteresting.

Mutual Information Based Pruning: According to [24], if two points x and y have probabilities $P(x)$ and $P(y)$, then their mutual information $I(x, y)$ is defined to be

$$I(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)}$$

Informally, mutual information compares the probability of observing x and y together (the joint probability) with the probabilities of observing x and y independently (chance). If there is a genuine association between x and y, then the joint probability $P(x, y)$ will be much larger than chance $P(x)P(y)$, and consequently $I(x, y) >> 0$. If there is no interesting relationship between x and y, then $P(x, y) \approx P(x)P(y)$, and thus, $I(x, y) \approx 0$. If x and y are in complementary distribution, then $P(x, y)$ will be much less than $P(x)$ $P(y)$, forcing $I(x, y) << 0$. Suppose that there is a rule $X \Rightarrow Y$, if $\text{supp}(X \Rightarrow Y) \approx \text{supp}(X) \times \text{supp}(Y)$, then it's likely that X and Y are independent and rule $X \Rightarrow Y$ is uninteresting, so we prune it.

Summarization: The following Rules

$A, B \Rightarrow C$ $A, C \Rightarrow B$ $B, C \Rightarrow A$

can be defined as association hyperedges, i.e., sets of items that are strongly predictive w.r.t. each other. The selection criterion is as follows: Given an item set with enough support, all rules are checked which can be formed using this set with all items appearing in the rule. For example, for the item set $\{A\ B\ C\}$, the rules $AB \Rightarrow C$, $AC \Rightarrow B$ and $BC \Rightarrow A$ would be considered. If the confidence of each rule is greater than the minimal confidence, the item set is selected. The confidence of the itemset is the average of the confidence of all rules;

For example: the two rules:

A ⇒ B (Supp.:16, Conf.: 93%)

B ⇒ A (Supp.:16, Conf.: 97%)

are association hyperedges. We can group them together and their confidences are the average of the confidences:

{A, B} (Supp.:16, Conf.: 95%)

After pruning and summarization, about 40% of the rule can be reduced.

4.5.Re-ranking

We propose a novel algorithm to re-rank the results using generalized association rule. In general, the pages in an association rule are accessed frequently together and mostly the content of the pages are relative, so we can use the association rule to improve the performance of site search.

Our algorithm is similar to DirectHit. Both of them

each consisting of several pages. These two topics are frequently co-visited. However, the standard association rule is unable to find the relationships between these two topics.

To overcome the above problems, we can utilize existing taxonomies, such as hierarchical clustering of content and site directory. These taxonomies contain some semantic information about the website. Traditional log mining algorithms only discover associations among leaf-level items in the taxonomy. While our method mine for association rules at different levels of abstraction. We wish access patterns may contain some interesting regularities at higher levels of abstraction. For example, the rule "$A \Rightarrow B$" may have insufficient support using standard association rule mining, but the rule "ancestor(A)$\Rightarrow B$" may pass the support requirement. That is because additional transactions may support "$A' \Rightarrow B$", where "ancestor (A) =ancestor (A')".

Following sub-sections will describe our method in detail.

4.1.Preprocessing

The starting and critical point for successful log mining is data pre-processing. The required tasks are data cleaning, user identification, and session identification.

An entry in Web server log contains the timestamp of a traversal from a source to a target page, the IP address of the originating host, the type of access (GET or POST) and other data. Many entries are considered uninteresting for mining and are removed. The filtering is application dependant. While in most cases accesses to embedded content such as images and scripts can be safely filtered out.

Table 1. Example of Log file

#	IP Address	Time/Date	Protocol	Request URI	Result
1	24.5.193.7	11/1/2001 12:00:30 AM	GET/HTTP 1.0	/~demmel/cs267/tsp/doc/taskq/node15.html	200
2	159.226.21.3	11/1/2001 13:03:08 AM	GET/HTTP 1.0	/~eran	200
3	169.229.90.77	11/1/2001 14:00:32 AM	GET/HTTP 1.0	/~chema/papers/dns/dns_report/node2.html	200
4	216.126.153.59	11/1/2001 15:00:00 AM	GET/HTTP 1.0	/~culler/cs258-s99/slides/lec05/sld026.htm	200

The remaining entries must be grouped by the visitors that performed them. An investigation on such approaches can be found in [19]. We currently assume that consecutive accesses from the same host during a certain time interval come from the same user.

Once we assess the originator of each entry, we group consecutive entries to a user session or "transaction". Different grouping criteria are modeled and compared in [19]. We support two criteria:

(1) A new session starts when the duration of the whole group of traversals exceeds a time threshold, similarly to [19].

(2) The elapsed time between two consecutive traversals exceeds a threshold.

After preprocessing, we can get a set of n pages, $P=\{p_1, p_2, \ldots, p_n\}$, and a set of m transactions, $T=\{t_1, t_2, \ldots, t_m\}$,where each $t_i \in T$ is a subset of P.

4.2.Site Taxonomy Building

By analyzing the Web pages, we found that pages are not randomly scattered. Many websites have a hierarchical organization of content, called page hierarchy. A page hierarchy is a partial order of Web pages, in which a leaf node represents a Web page corresponding to a file in the website. A non-leaf node in a page hierarchy represents a Web directory in the website.

We construct taxonomy Γ using the hierarchy above; the taxonomy is initialized with only the root which represents the top level of the website. For each URL in the URL list generated, if it does not exist in the taxonomy, a node for the page is created. Next we parse the URL and for each prefix, which is a directory, we create an ancestor node if it doesn't exist in the taxonomy. Take the URL http://www.cs.berkeley.edu/~jordan /courses/index.html as an example, its first prefix http://www.cs.berkeley.edu/~jordan/courses/ and its second prefix http://www.cs.berkeley.edu/~jordan/ may be created and a link is added between itself and its first prefix, between its first prefix and its second prefix, and between its second prefix and the root. The Figure 3 is the result of the website http://www.cs.berkeley.edu:

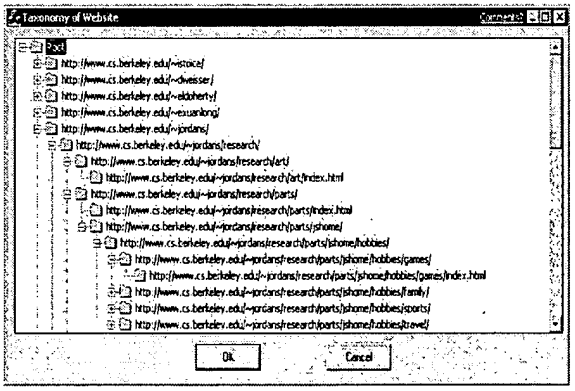

Figure 3. A website taxonomy

4.3.Mining Algorithm

To support taxonomies, one can use algorithms *apriori* [17] for mining standard association rules by considering "extended transaction" that contains not only the items in transactions but also their ancestors. To make this process efficient, certain optimizations are done to restrict the number of itemsets that need to be counted at

transactions in D that contain X also contain Y. We say that a rule $X \Rightarrow Y$ holds in transaction set D with *support* s% if s% of transactions in D contain both X and Y.

Returning to the above example, suppose we find that in 90% of transactions if user accessed page A and page B, they also accessed page C. Moreover, say that 5% of transactions include all three items. Consequently, the confidence of the rule is 90% and the support is 5%.

Association Rule Mining Problem: Given a set of transactions D, the problem of mining association rules is to generate all association rules that have support s% at least as great as some user-specified minimum support s_{min}% and confidence c% at least as great as some user-specified minimum confidence c_{min}%.

Several algorithms have been presented in the literature [5][16][17] for finding all such association rules. Many of them are variations of the *Apriori* algorithm. Apriori algorithm has two phases: (1) it finds all itemsets that have support above the minimum support; and (2) it uses these itemsets to generate all rules whose confidence are above the minimum confidence.

Taxonomy: By taxonomy, we mean "is-a hierarchy" (as shown in Figure 1) where a node's descendents represent specializations of that node. For example, the taxonomy in Figure 1 says that Jacket is-a Outerwear and Outerwear is-a Clothes.

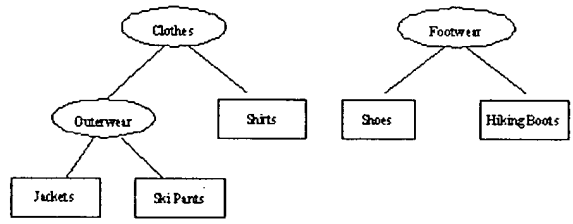

Figure 1. Taxonomy Structure

Formally speaking, we model one or more taxonomies as a directed acyclic graph Γ on the items $I = \{i_1, i_2, \ldots, i_m\}$. Edges in Γ denote "is-a" relationships among items. Specifically, an edge from c up to p in Γ indicates that p is the parent of c and c is a particular kind of p (or in other words, that p is a generalization of c).

Generalized association rules: Generalized association rules [22] improve upon standard association rules by incorporating a taxonomy Γ. In particular, a generalized association rule is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \emptyset$, and no item in Y is an ancestor of any item in X. The reason for the latter requirement is that any rule of the form "$x \Rightarrow$ ancestor(x)" is true with 100% confidence and consequently redundant.

Now, we say that a generalized association rule $X \Rightarrow Y$ holds in transaction set D with confidence c% if c% of transactions in D that contain X or a descendent of X also contain Y or a descendent of Y. Moreover, itemset X has support s% in transaction set D if s% of transactions in D contain Z or a descendent of Z.

4.GENERALIZED ASSOCIATION RULE MINING FOR SITE SEARCH

We developed a general log mining model to discover generalized association rules for site search. This model is illustrated in Figure 2.

As can be seen from Figure 2, the model uses the log files and a taxonomy as the input, and output a set of association rules. When a user poses a query, a full-text search engine finds all the Web pages that match the query words. These Web pages are then re-ranked by these association rules.

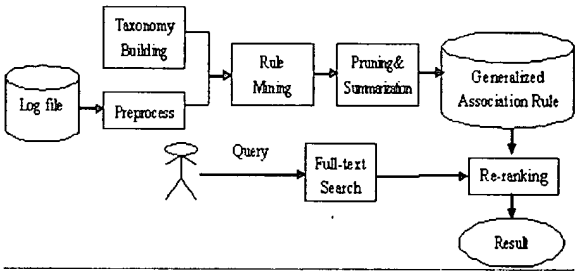

Figure 2. Flow of site search using generalized association rule

Our method mines for generalized association rule instead of standard association rule to tackle the data diversity problems. Earlier works on association rule mining [1][3][7][21] only mine for relationships among distinct Web pages, which lead to three main problems:

- First, our statistics on a real Web log show that most of the pages have low hit rate. As can be seen from Figure 5, about 80% of the pages are visited less than 10 times. If we use standard association rule, those pages are always ignored. However, in most of cases, these pages may contain latent information about the user's access patterns which can't be discovered using standard association rule mining.
- Second, a website usually contains thousands, even millions of pages. It is not easy to find those users who access some common pages. This is because of the diversity of the users. Moreover, this also leads to the difficulty of finding the same access pattern.
- Third, there are some latent semantic topics in a website. For example, there exist two topics in Berkeley CS's website: AI and machine learning,

the time, duration and URL. We can obtain from these logs each page's access frequency and the traversal patterns of information finding. They reflect Web pages' importance in the users' point of view and associations among Web pages, which can be used to improve the performance of site search.

Generally, the process of discovering useful patterns from Web logs is called *log mining* [21] (or *usage mining*). Log mining includes straightforward statistics methods, such as page access frequency, as well as more sophisticated forms of analysis, such as association rule mining, sequential pattern mining, clustering, and etc. In this paper, we are particularly interested in association rule mining [22]. To our best knowledge, there is no effort devoted to improving the site-search performance through association rule mining. This paper will present such a method in detail.

A normal association rule mining algorithm may fail to discover significant rules due to the data diversity problem. Instead, we use generalized association rule mining to utilize a predefined taxonomy and extract significant association rules at different abstract level of the taxonomy. We designed an algorithm which is based on the FP-growth algorithm [5] to mine for generalized association rules efficiently. These association rules are subsequently pruned and applied to page re-ranking.

Here we summarize the contributions of our work: (1) proposal of utilizing Web page access logs to improve site search, (2) an efficient tool that do log preprocessing, taxonomy generating, generalized association rule mining, rule pruning, and search result re-ranking, and (3) experiment of our method on Berkeley CS website, in which the result outperforms keyword-based method by 15% and DirectHit by 13% respectively.

The organization of the paper is as follows. In the next section, we review the related work. In Section 3, some basic terms and algorithms used in log mining are given. In section 4, we describe the mining algorithm for generalized association rules and the re-ranking mechanism in site search. In Section 5, we present our experimental results related to this new method, and conclude in Section 6.

2.RELATED WORK

Because of the importance of site search, many works have been done to improve the accuracy of site search. In the mean time, association rule mining has also received much attention on various areas, such as recommendation system, collaborative filtering, site modification, etc. These two research topics were formerly developed independently. In this section, we discuss some conventional approaches of site search and applications of association rules mining.

Traditionally, most site search engines merely use "full text search" technology, which retrieves a large amount of documents containing the same keywords provided by the user. For example, the Cha-Cha system [12] is a browsing assistant. It organizes Web search results in a way that can reflect the underlying structure of the intranet. In the system, an "outline" or "table of contents" is created by first recording the shortest paths in hyperlinks from root pages to every page within the Web intranet. After the user issues a query, those shortest paths are dynamically combined to form a hierarchical outline of the context in which the search result occurs. Therefore, Cha-Cha system's goal is to provide well-organized search results. It cannot save users' time when the relevance of search result is poor.

M. Levence [13] described a semi-automatic navigation system NavigationZone. It builds trails of information, i.e. sequences of linked pages, which are relevant to the posed query and provides a good starting point for users to initiate a navigation session. Nevertheless, the system does not further use the associated relation of the pages to help user navigate and re-rank the search result.

In the following, we briefly discuss the idea of association rule mining and its applications. Since its introduction in 1993, association rule mining has received a great deal of attention. Today it is still one of the most popular pattern-discovery methods in data mining.

Earlier works on association rule based log mining focused on several applications. They have been used to mine for path traversal patterns and to facilitate the best design and organization of Web pages [20][10][11]. Some recommender systems [25] have been developed to recommend Web pages. They used the *Apriori* algorithm to mine for association rules over users' navigation histories. In other applications such as access prediction [26] and page pre-fetching [15], association rule mining is also frequently used.

3.PRELIMINARIES

In this section, we list some basic terms and algorithms used in association rule based log mining.

Association Rules: Let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of *items*. An *association rule* is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \emptyset$.

For example, suppose users who accessed page a and page b also tend to access page c. The corresponding association rule is "$A \wedge B \Rightarrow C$". The antecedent of the rule X consists of A and B, and the consequent Y consists of C.

Confidence and Support: Let D be a set of transactions, where each transaction $T \in D$ is a set of items (itemset) such that $T \subseteq I$. We say that a rule $X \Rightarrow Y$ holds in transaction set D with *confidence* c% if c% of

Log Mining to Improve the Performance of Site Search

Gui-Rong Xue[1] Hua-Jun Zeng[2] Zheng Chen[2] Wei-Ying Ma[2] Chao-Jun Lu[1]

[1]Computer Science and Engineering
Shanghai Jiao-Tong University
Shanghai 200030, P.R.China
grxue@sjtu.edu.cn, cj-lu@cs.sjtu.edu.cn

[2]Microsoft Research Asia
49 Zhichun Road, Beijing 100080, P.R.China
{i-hjzeng, zhengc, wyma}@microsoft.com

ABSTRACT

Despite of the popularity of global search engines, people still suffer from low accuracy of site search. The primary reason lies in the difference of link structures and data scale between global Web and website, which leads to failures of traditional re-ranking methods such as HITS, PageRank and DirectHit. This paper proposes a novel re-ranking method based on user logs within websites. With the help of website taxonomy, we mine for generalized association rules and abstract access patterns of different levels. Mining results are subsequently used to re-rank the retrieved pages. One of the advantages of our mining algorithm is that it resolves the diversity problem of user's access behavior and discovers general patterns. Experiment shows that the proposed method outperforms keyword-based method by 15% and DirectHit by 13% respectively.

1.INTRUDUCTION

Global search engines such as Google, AltaVista, and Lycos have been a great help for users to find desired information on the ever growing Web. Given clear and unambiguous queries, they can return desired results most of the time. However, this is not always the case. As pointed out in [9], users often pose unclear and general queries to find appropriate websites as good starting points. Once at a site, the user has a choice of following hyperlinks or using site search to get more specific information. Due to the low-efficiency of following hyperlinks, there is a tremendous need for site search techniques. As reported in a recent Forrester survey [14], website managers also consider search to be a critical factor of their sites' functionality. In addition, global search engines cannot index content within dynamic websites or intranet, where site search is the only way for the users to find information.

Site search can be simply defined as the search functionality specific to one website. However, unlike global search engines, site search engines are notoriously problematic at present. In [14], Forrester tested site search facilities of 50 websites, but none of them returned satisfying results. For example, they often didn't find the content that best matches what the user wanted; they rarely put all the best content on the first page of results; they typically return more irrelevant results than useful ones. What are the reasons of these failures?

Most site search engines merely use "full text search" method which retrieves a large amount of documents containing the same keywords inputted by the user. Due to shortness of query words and poor ranking mechanism, it is a time consuming job for the users to go through the results to find out their really desired information.

Many techniques, which are very successful in Web search, seem directly applicable in site search, such as link analysis [6][23] and clickthru-based ranking [3]. But neither of them can work well, for the following reasons. First, the link analysis techniques, such as HITS [6] and PageRank [23], use hyperlinks among Web pages to rank pages, where pages with more reference get higher ranking score. However the link information within a website is not strong enough to reflect the page's score. Thus the most important Web pages are not necessarily the highest referenced page; high referenced pages are often the home page, index pages and help pages which are not really wanted by users. The failure of applying link analysis to Web TREC datasets [2] also demonstrate that link analysis doesn't work for a sub-space of Web.

Second, clickthru-based ranking methods such as DirectHit [3] are also problematic to be used in site search engines. According to a particular query, DirectHit utilizes previous session logs of the same query to return pages that most users visited. To get a statistically significant result, it is only applied to a small set of popular queries. Because of the lack of previous query sessions and the diversity of user's access patterns, DirectHit doesn't work for site search either.

In this paper, we propose a novel re-ranking method based on site logs. Every website keeps a set of access logs, which embody browsing behaviors of its users and